Figure 1

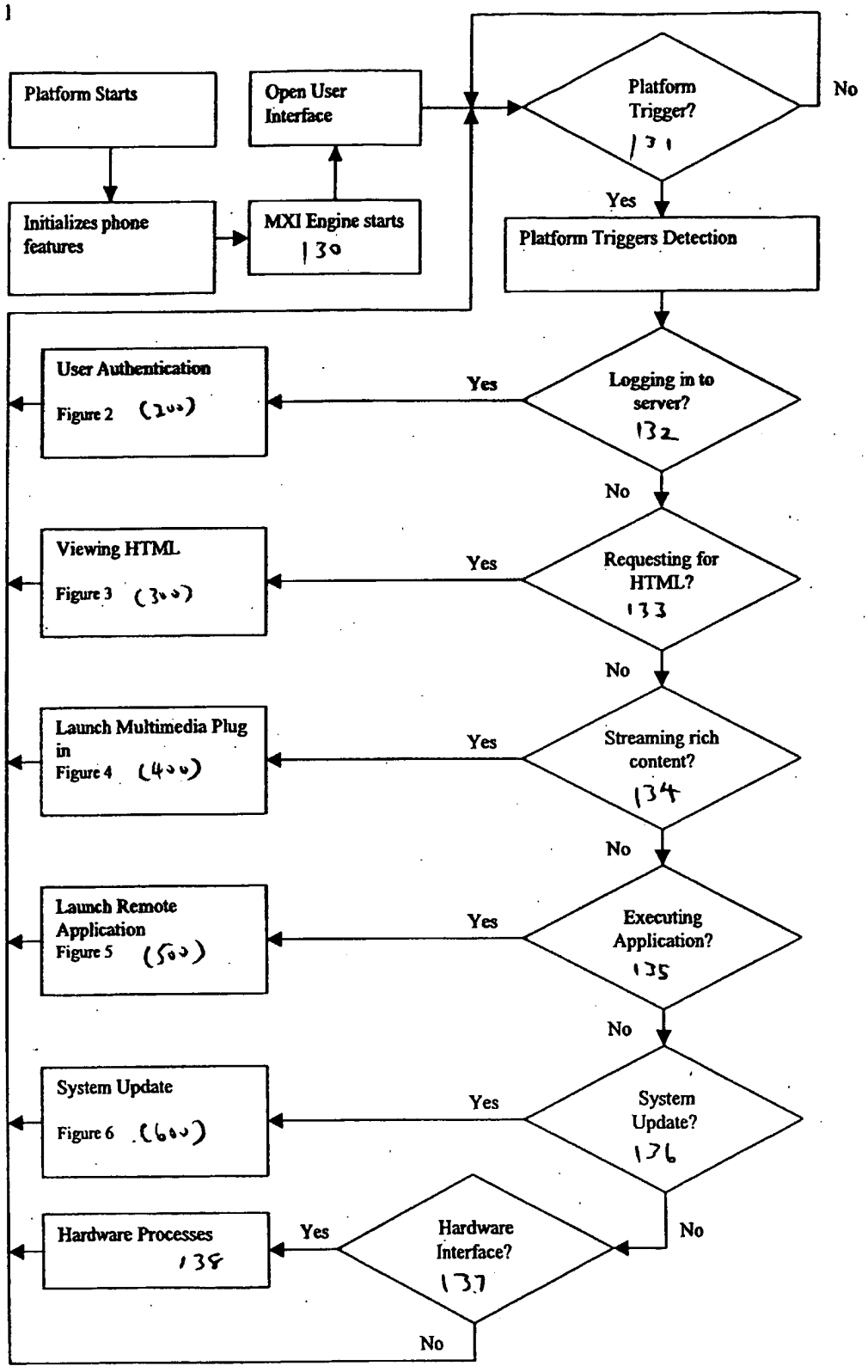

Platform Starts
Open User Interface
Platform Trigger?
131
No
Yes
Initializes phone features
MXI Engine starts
130
Platform Triggers Detection
User Authentication
Figure 2 (200)
Yes
Logging in to server?
132
No
Viewing HTML
Figure 3 (300)
Yes
Requesting for HTML?
133
No
Launch Multimedia Plug in
Figure 4 (400)
Yes
Streaming rich content?
134
No
Launch Remote Application
Figure 5 (500)
Yes
Executing Application?
135
No
System Update
Figure 6 (600)
Yes
System Update?
136
No
Hardware Processes
138
Yes
Hardware Interface?
137
No

Figure 2

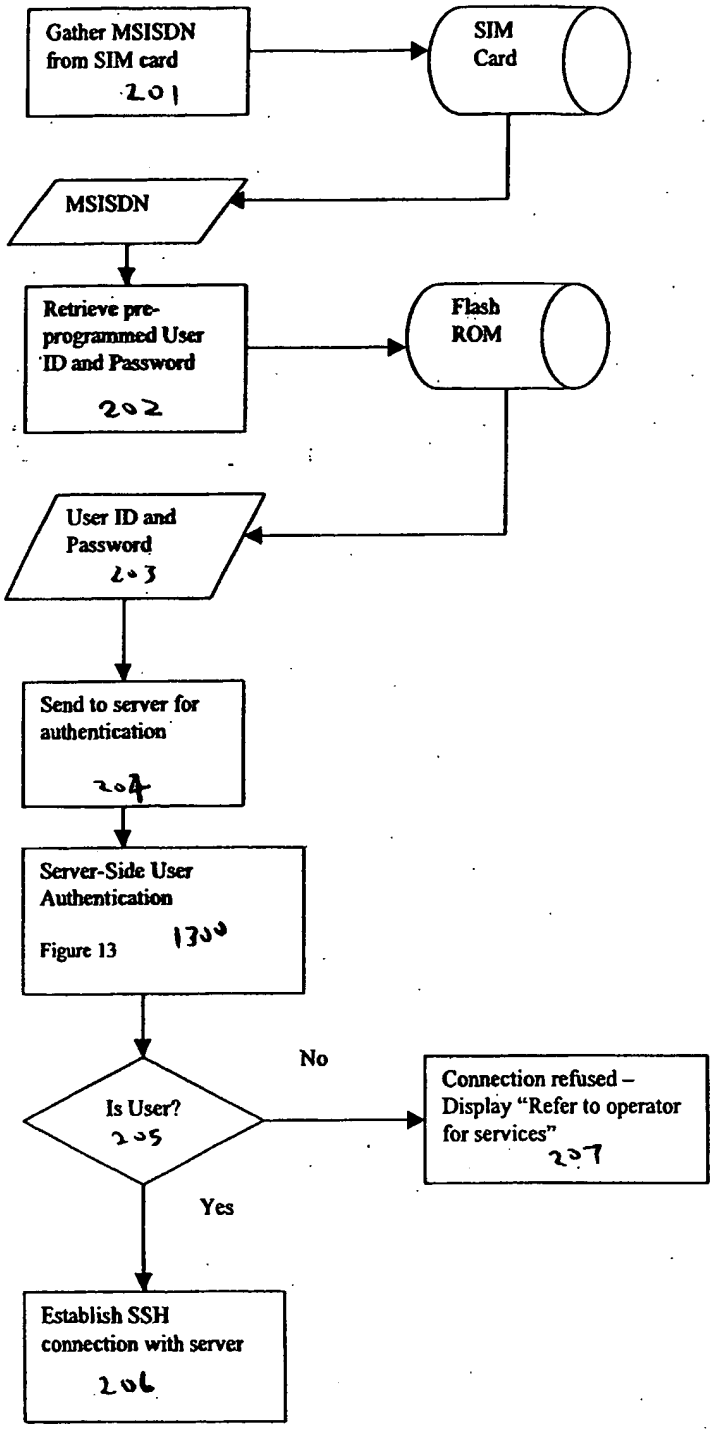

Gather MSISDN from SIM card
201
SIM Card
MSISDN
Retrieve pre-programmed User ID and Password
202
Flash ROM
User ID and Password
203
Send to server for authentication
204
Server-Side User Authentication
Figure 13
1300
Is User?
205
No
Connection refused – Display "Refer to operator for services"
207
Yes
Establish SSH connection with server
206

Figure 3

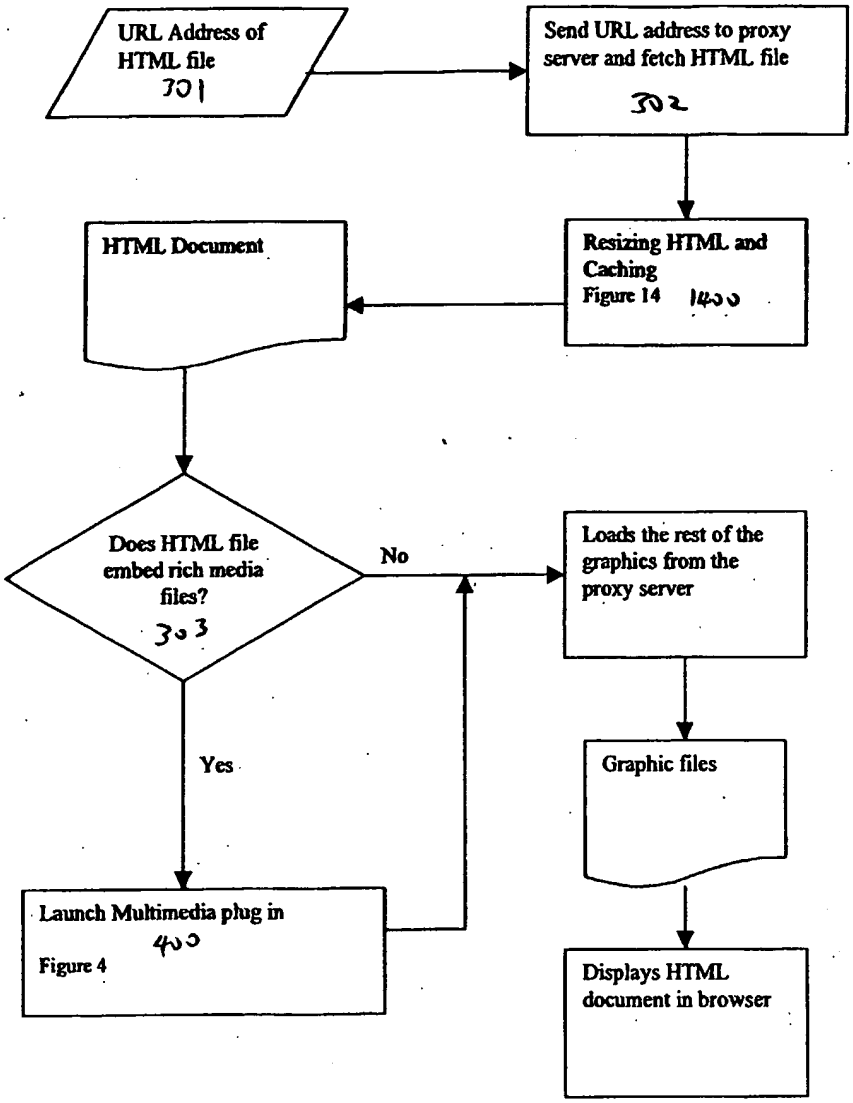

URL Address of HTML file
301
Send URL address to proxy server and fetch HTML file
302
HTML Document
Resizing HTML and Caching
Figure 14
1400
Does HTML file embed rich media files?
303
No
Loads the rest of the graphics from the proxy server
Yes
Graphic files
Launch Multimedia plug in
400
Figure 4
Displays HTML document in browser

Figure 4

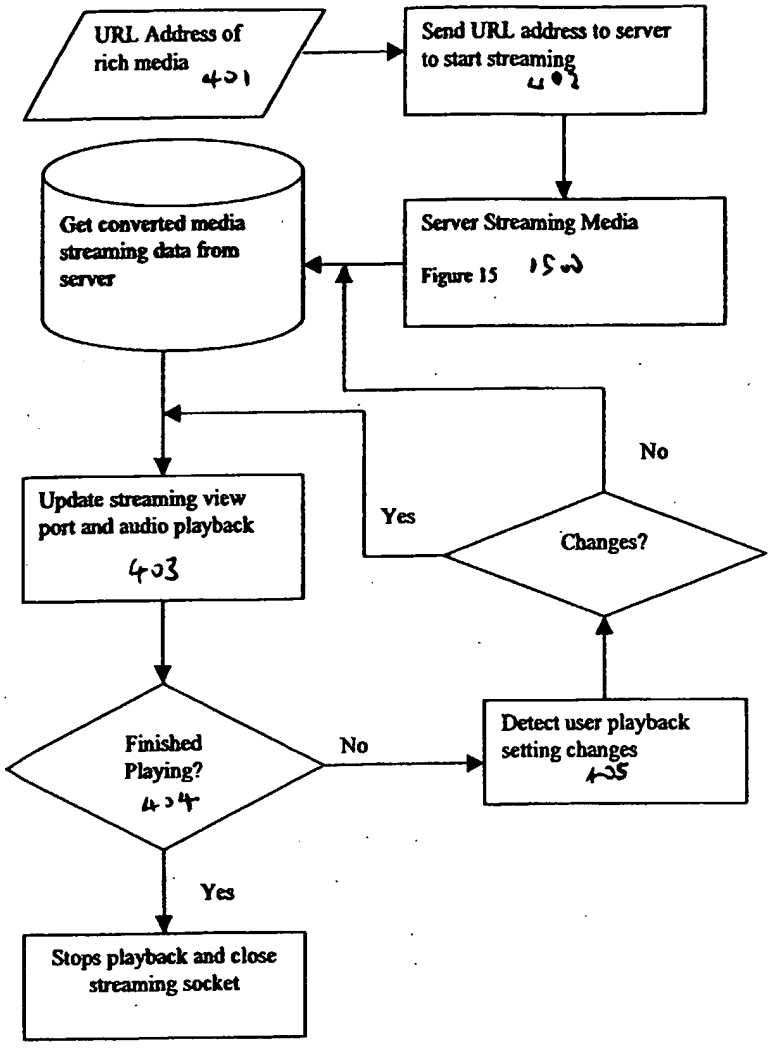

URL Address of rich media
401
Send URL address to server to start streaming
402
Get converted media streaming data from server
Server Streaming Media
Figure 15
150
Update streaming view port and audio playback
403
Yes
No
Changes?
Finished Playing?
404
No
Detect user playback setting changes
405
Yes
Stops playback and close streaming socket

Figure 5

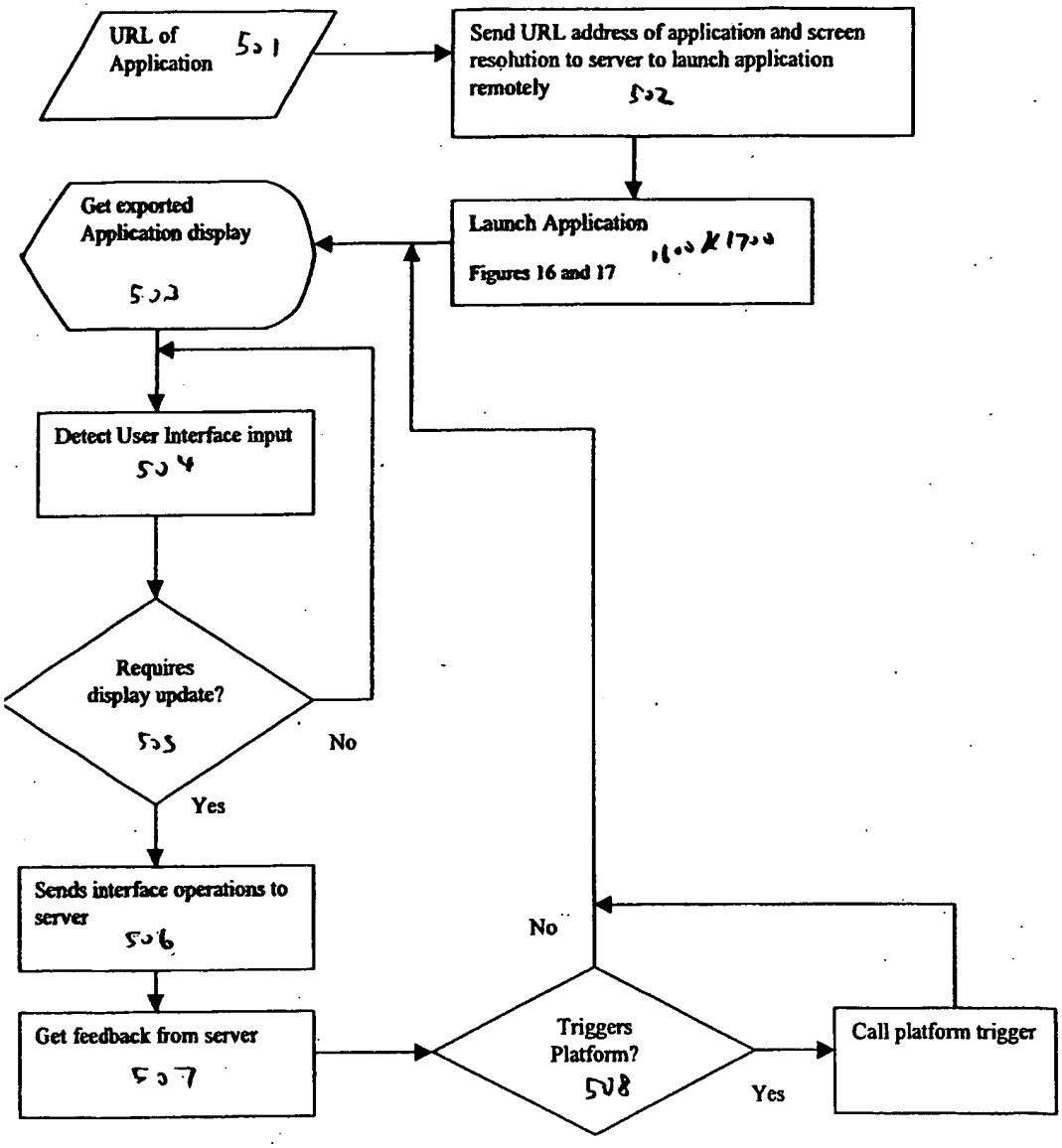

URL of Application 501
Send URL address of application and screen resolution to server to launch application remotely 502
Launch Application
Figures 16 and 17
Get exported Application display 503
Detect User Interface input 504
Requires display update? 505
No
Yes
Sends interface operations to server 506
Get feedback from server 507
Triggers Platform? 508
No
Yes
Call platform trigger

Figure 6

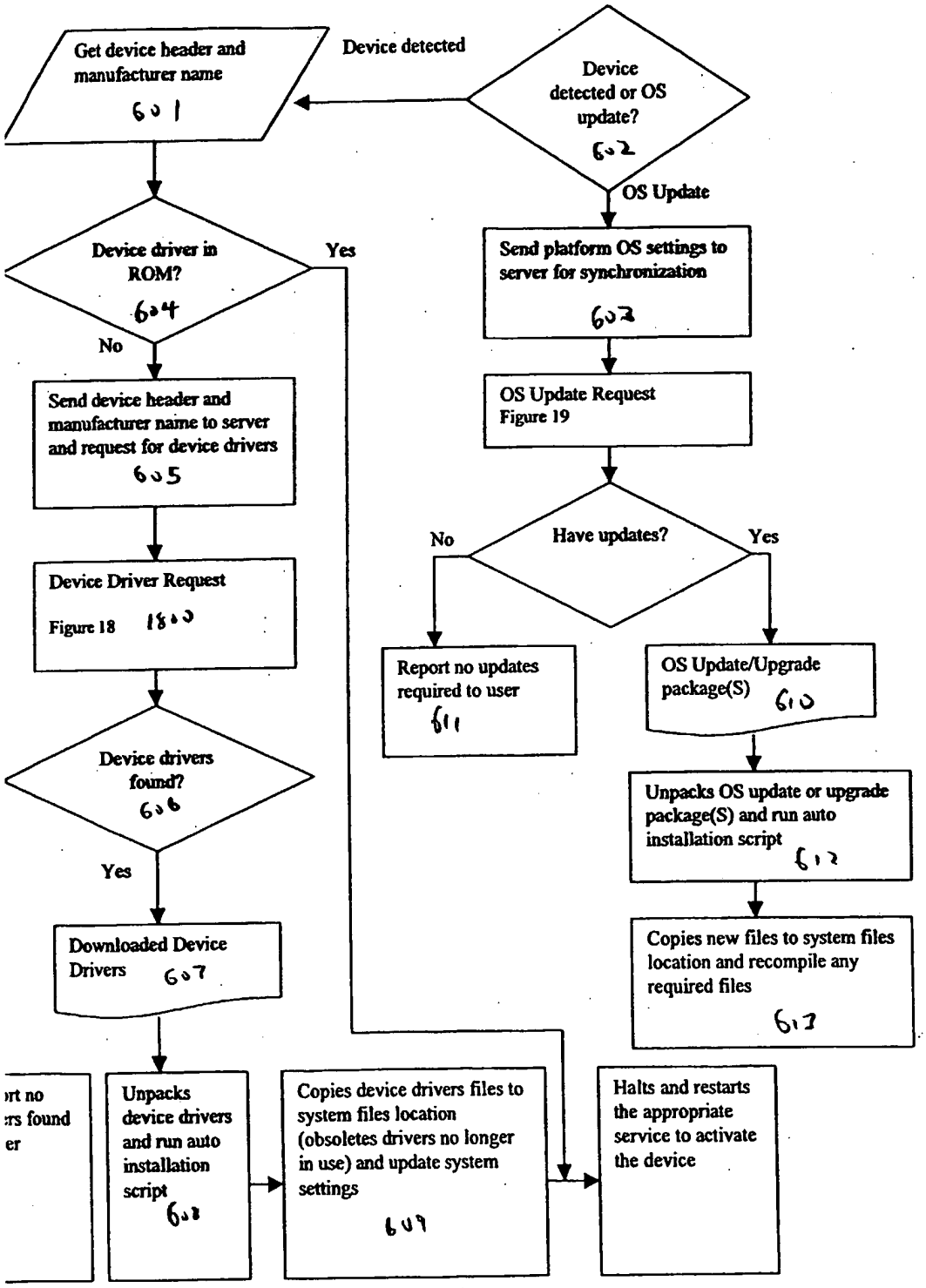

Get device header and manufacturer name
601
Device detected
Device detected or OS update?
602
OS Update
Device driver in ROM?
604
Yes
No
Send platform OS settings to server for synchronization
603
OS Update Request
Figure 19
Send device header and manufacturer name to server and request for device drivers
605
Have updates?
No
Yes
Device Driver Request
Figure 18
1800
Report no updates required to user
611
OS Update/Upgrade package(S)
610
Device drivers found?
606
Unpacks OS update or upgrade package(S) and run auto installation script
612
Yes
Copies new files to system files location and recompile any required files
613
Downloaded Device Drivers
607
ort no
ers found
er
Unpacks device drivers and run auto installation script
608
Copies device drivers files to system files location (obsoletes drivers no longer in use) and update system settings
609
Halts and restarts the appropriate service to activate the device

Figure 7

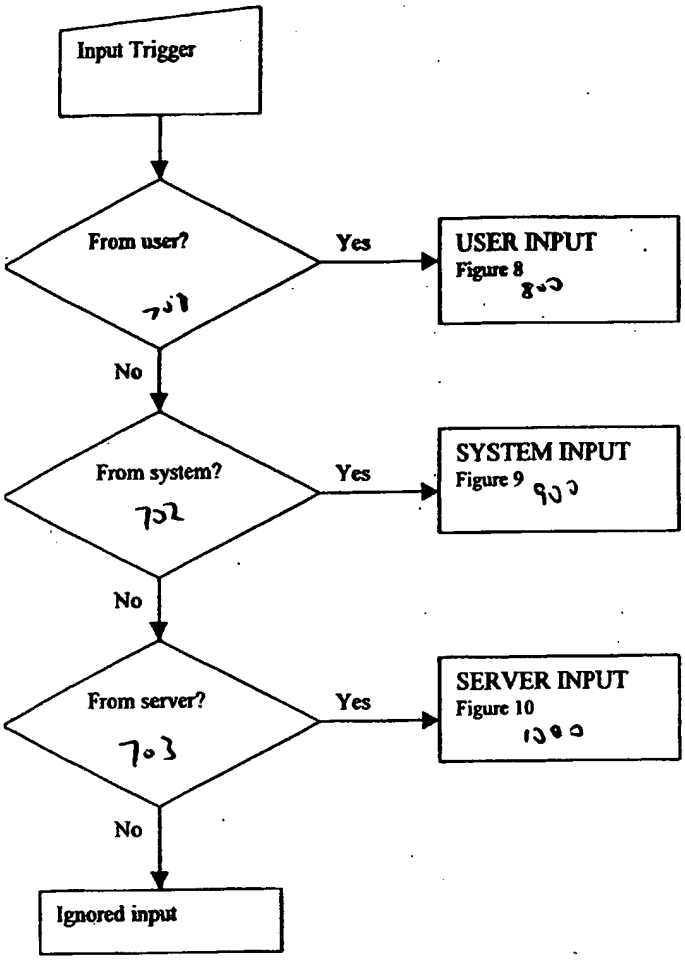

Input Trigger
From user?
701
Yes
USER INPUT
Figure 8
800
No
From system?
702
Yes
SYSTEM INPUT
Figure 9
900
No
From server?
703
Yes
SERVER INPUT
Figure 10
1000
No
Ignored input

Figure 8

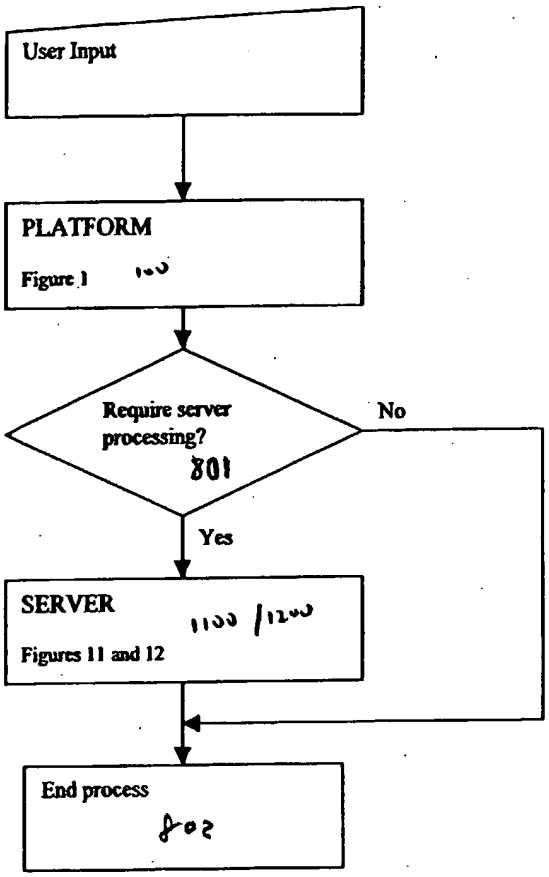

User Input
PLATFORM
Figure 1
Require server processing?
801
No
Yes
SERVER
Figures 11 and 12
End process
802

Figure 9

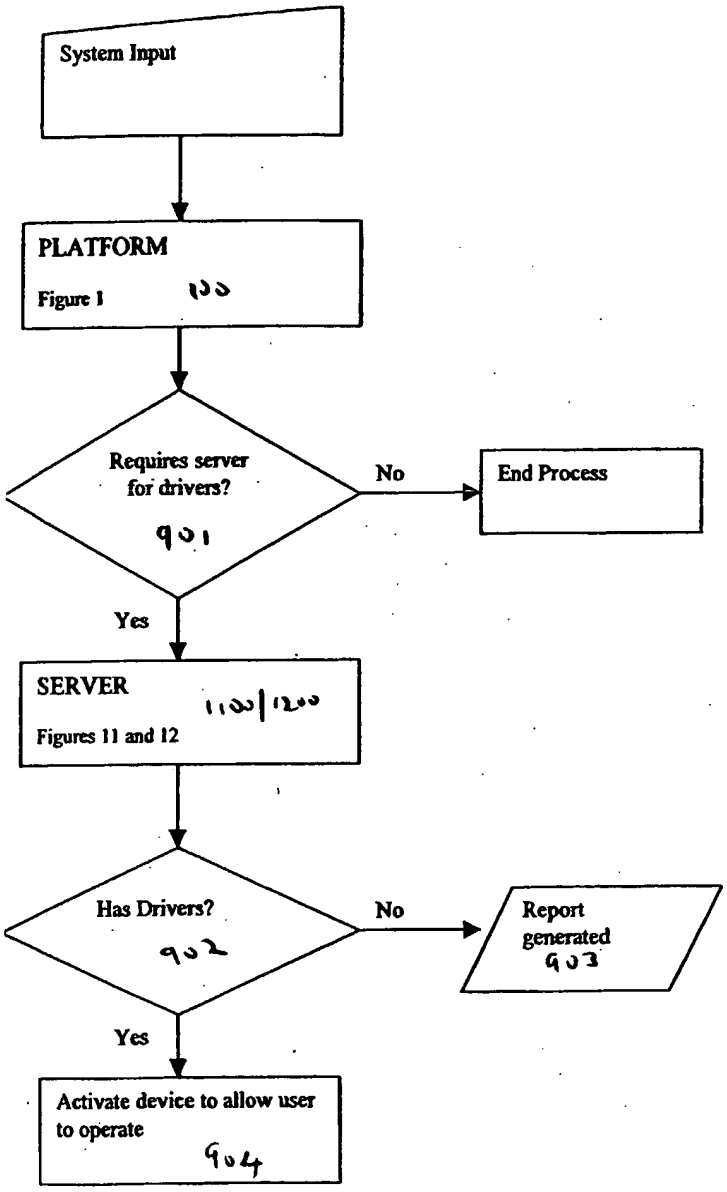

System Input
PLATFORM
Figure 1
Requires server for drivers?
901
No
End Process
Yes
SERVER
Figures 11 and 12
Has Drivers?
902
No
Report generated
903
Yes
Activate device to allow user to operate
904

Figure 10

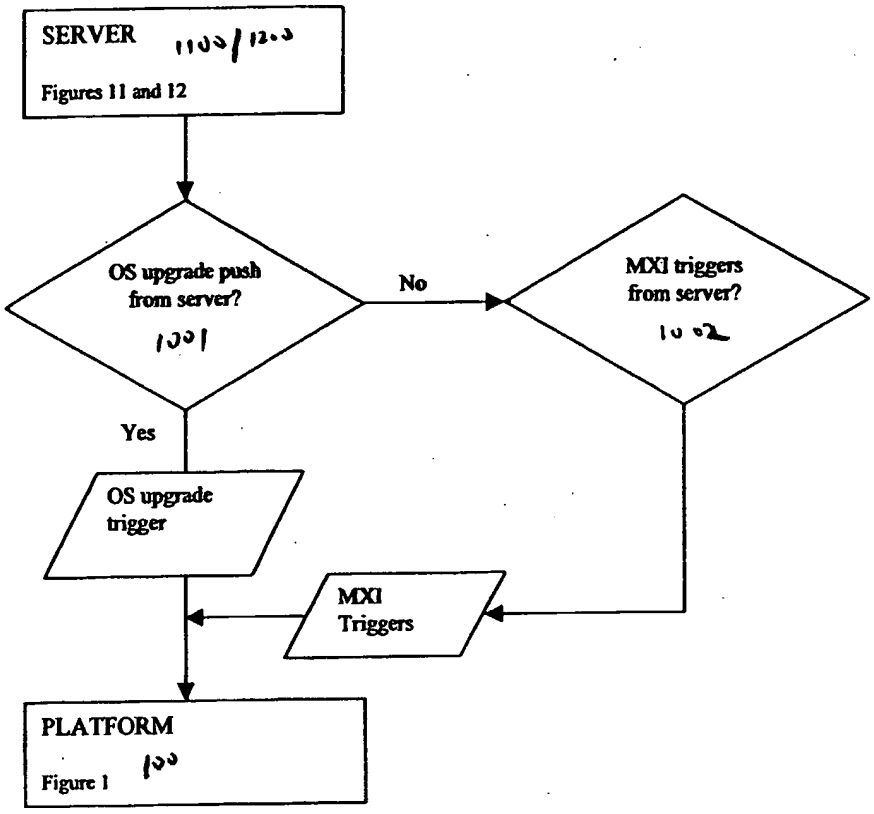

SERVER
1100/1200
Figures 11 and 12
OS upgrade push from server?
1001
No
MXI triggers from server?
1002
Yes
OS upgrade trigger
MXI Triggers
PLATFORM
100
Figure 1

Figure 11

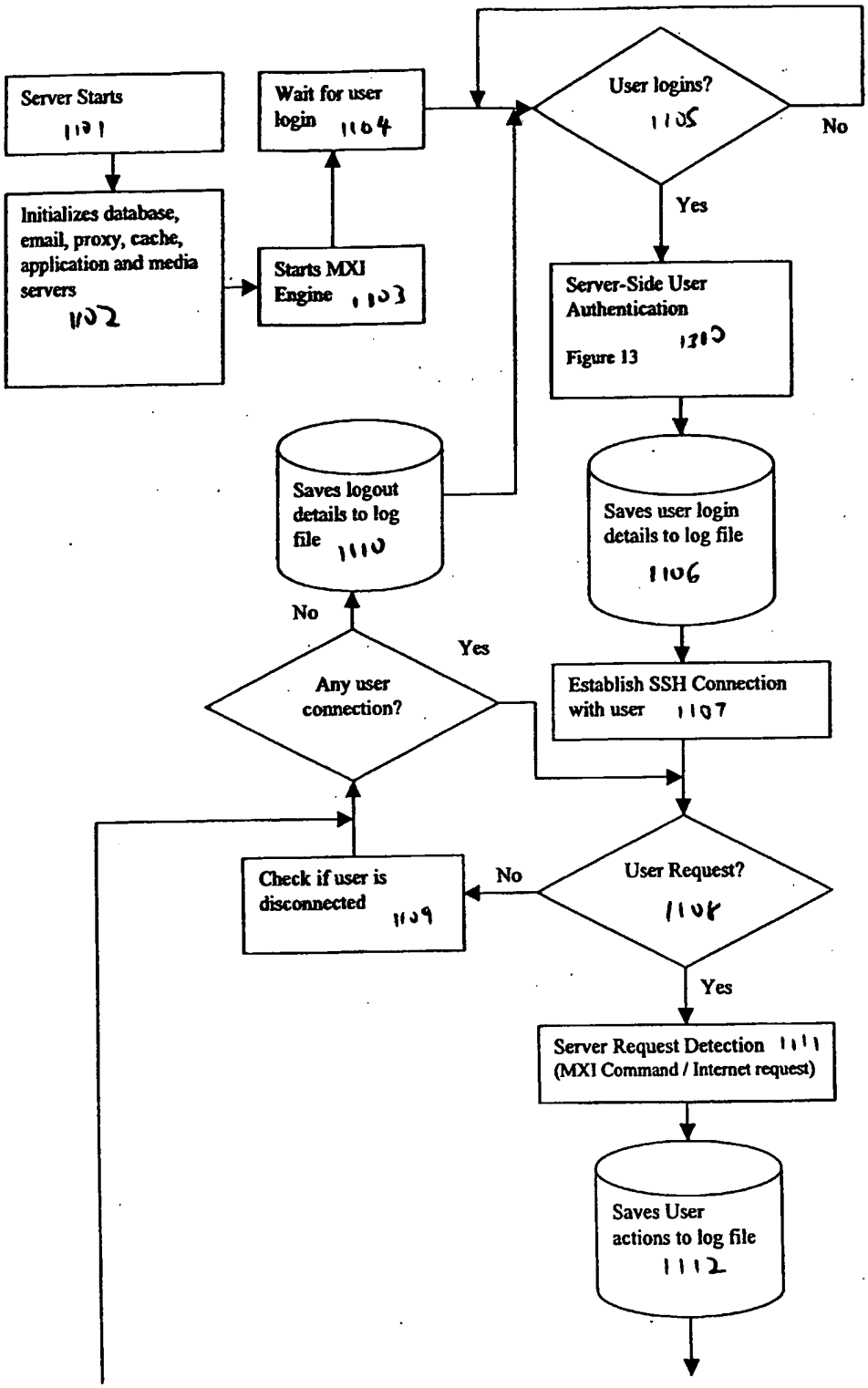

Server Starts
1101
Initializes database, email, proxy, cache, application and media servers
1102
Starts MXI Engine 1103
Wait for user login 1104
User logins? 1105
No
Yes
Server-Side User Authentication
1210
Figure 13
Saves user login details to log file
1106
Establish SSH Connection with user 1107
User Request? 1108
No
Check if user is disconnected 1109
Any user connection?
No
Yes
Saves logout details to log file 1110
Yes
Server Request Detection 1111
(MXI Command / Internet request)
Saves User actions to log file
1112

Figure 12

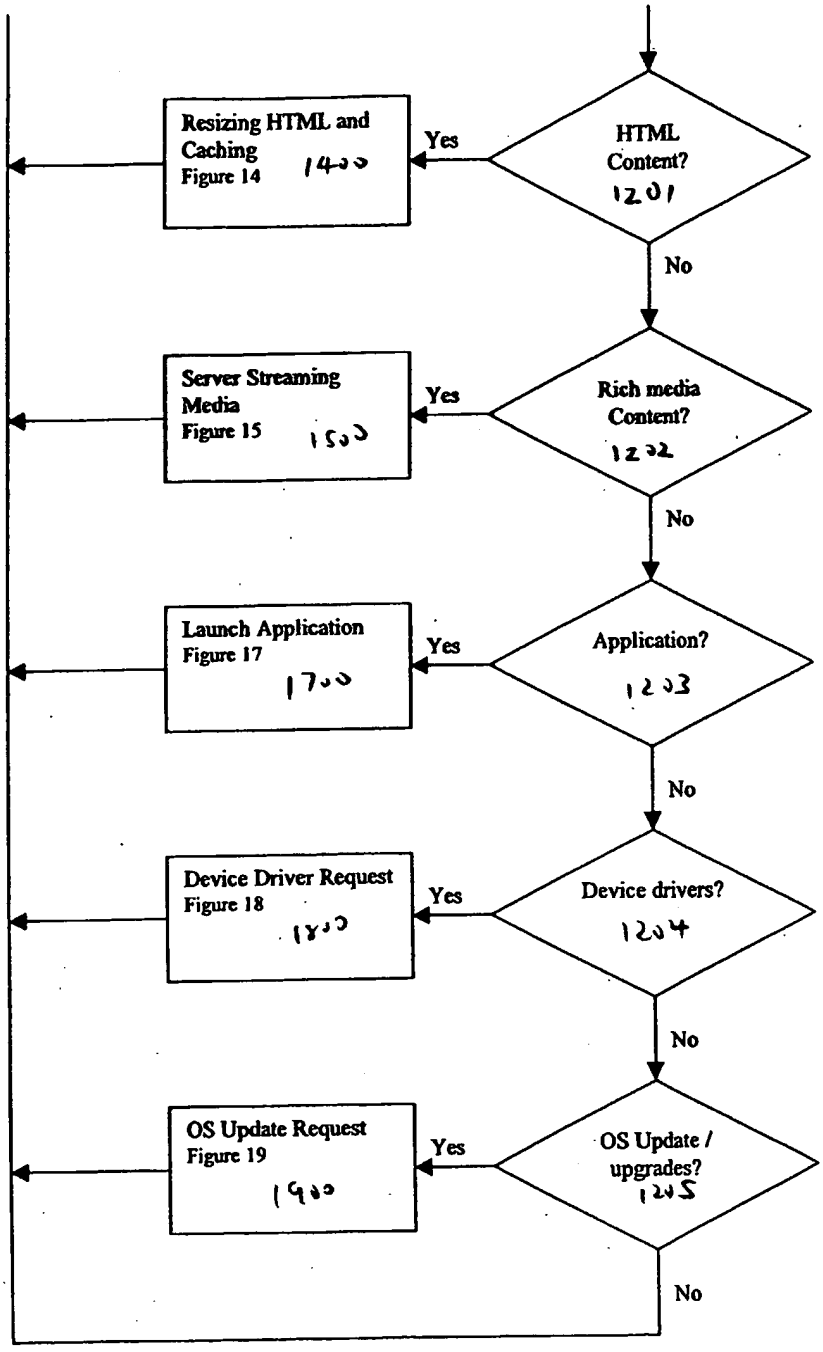

HTML Content?
1201
Yes
Resizing HTML and Caching
Figure 14
1400
No
Rich media Content?
1202
Yes
Server Streaming Media
Figure 15
1500
No
Application?
1203
Yes
Launch Application
Figure 17
1700
No
Device drivers?
1204
Yes
Device Driver Request
Figure 18
1800
No
OS Update / upgrades?
1205
Yes
OS Update Request
Figure 19
1900
No

Figure 13

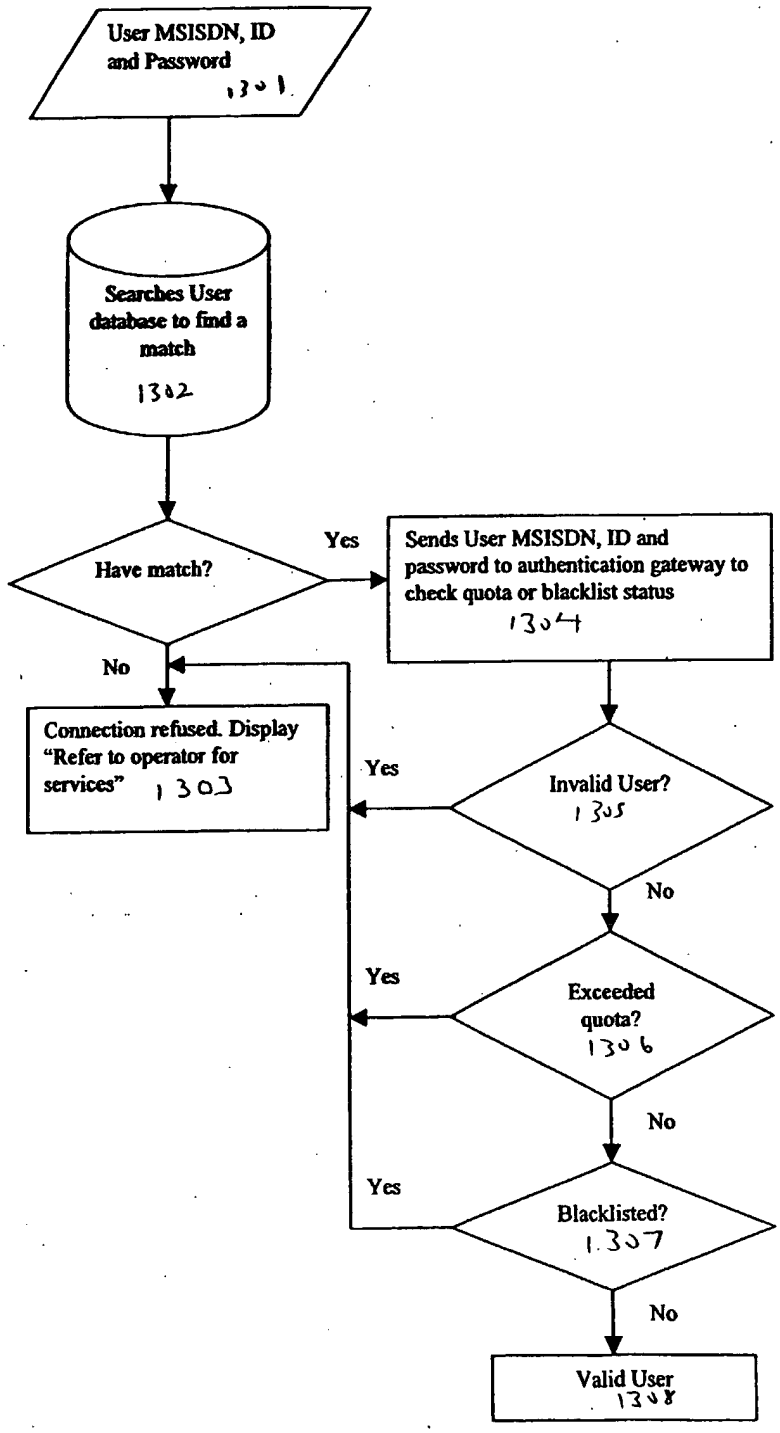

User MSISDN, ID and Password
1301
Searches User database to find a match
1302
Have match?
Yes
Sends User MSISDN, ID and password to authentication gateway to check quota or blacklist status
1304
No
Connection refused. Display "Refer to operator for services"
1303
Invalid User?
1305
Yes
No
Exceeded quota?
1306
Yes
No
Blacklisted?
1307
Yes
No
Valid User
1308

Figure 14

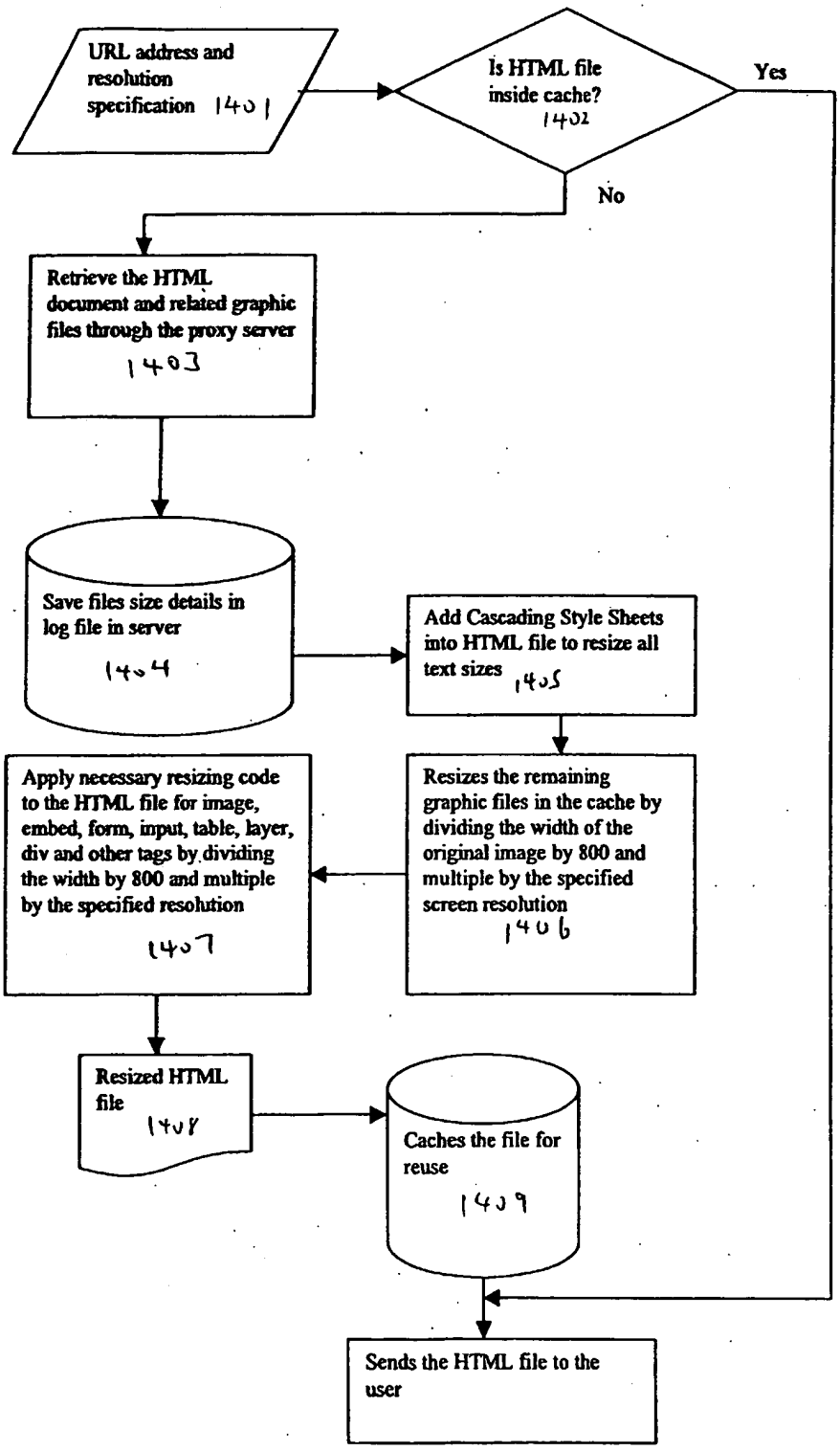

URL address and resolution specification 1401
Is HTML file inside cache? 1402
Yes
No
Retrieve the HTML document and related graphic files through the proxy server 1403
Save files size details in log file in server 1404
Add Cascading Style Sheets into HTML file to resize all text sizes 1405
Apply necessary resizing code to the HTML file for image, embed, form, input, table, layer, div and other tags by dividing the width by 800 and multiple by the specified resolution 1407
Resizes the remaining graphic files in the cache by dividing the width of the original image by 800 and multiple by the specified screen resolution 1406
Resized HTML file 1408
Caches the file for reuse 1409
Sends the HTML file to the user

Figure 15

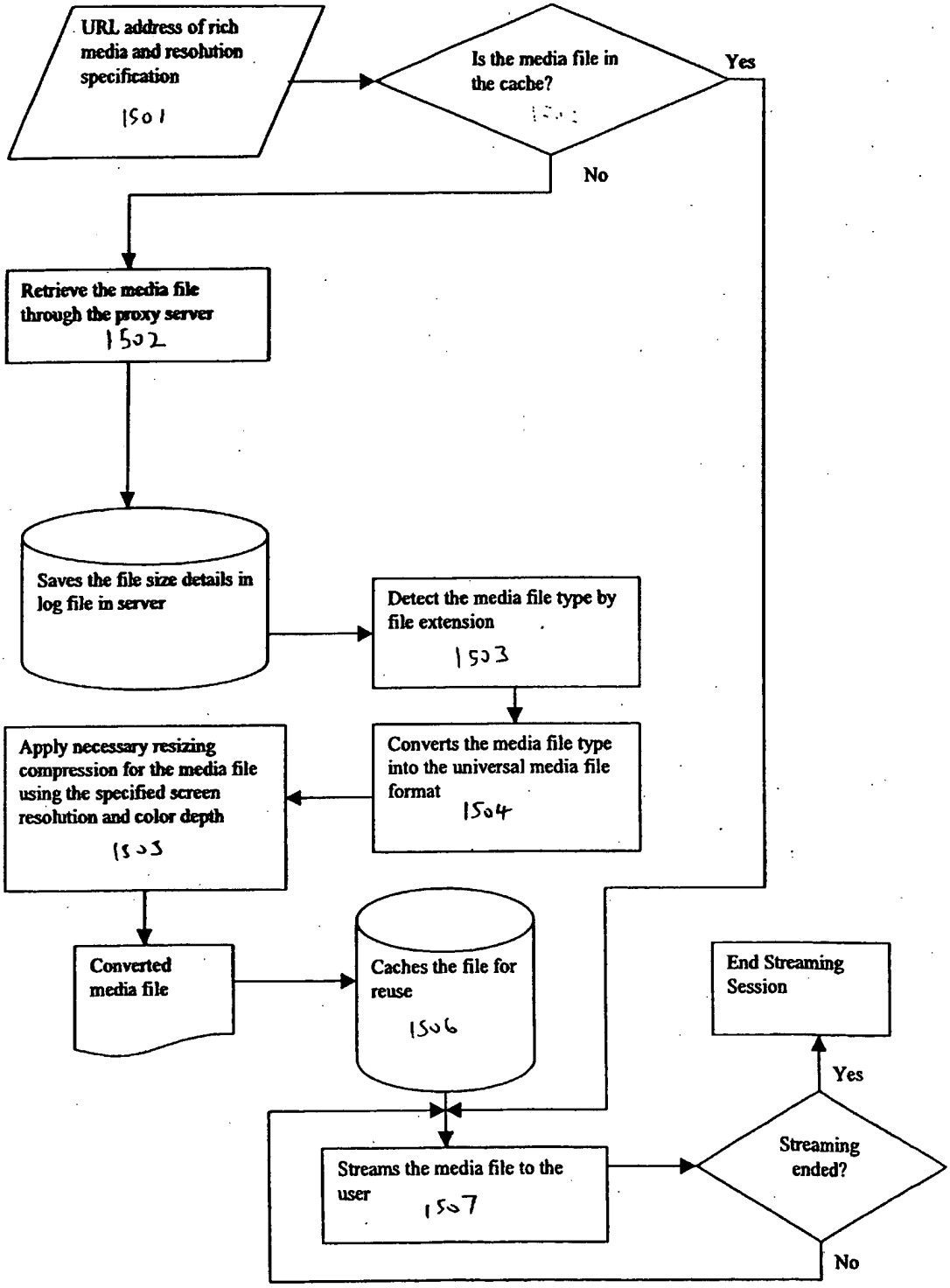

URL address of rich media and resolution specification
1501
Is the media file in the cache?
Yes
No
Retrieve the media file through the proxy server
1502
Saves the file size details in log file in server
Detect the media file type by file extension
1503
Converts the media file type into the universal media file format
1504
Apply necessary resizing compression for the media file using the specified screen resolution and color depth
1505
Converted media file
Caches the file for reuse
1506
End Streaming Session
Yes
Streaming ended?
Streams the media file to the user
1507
No

Figure 16

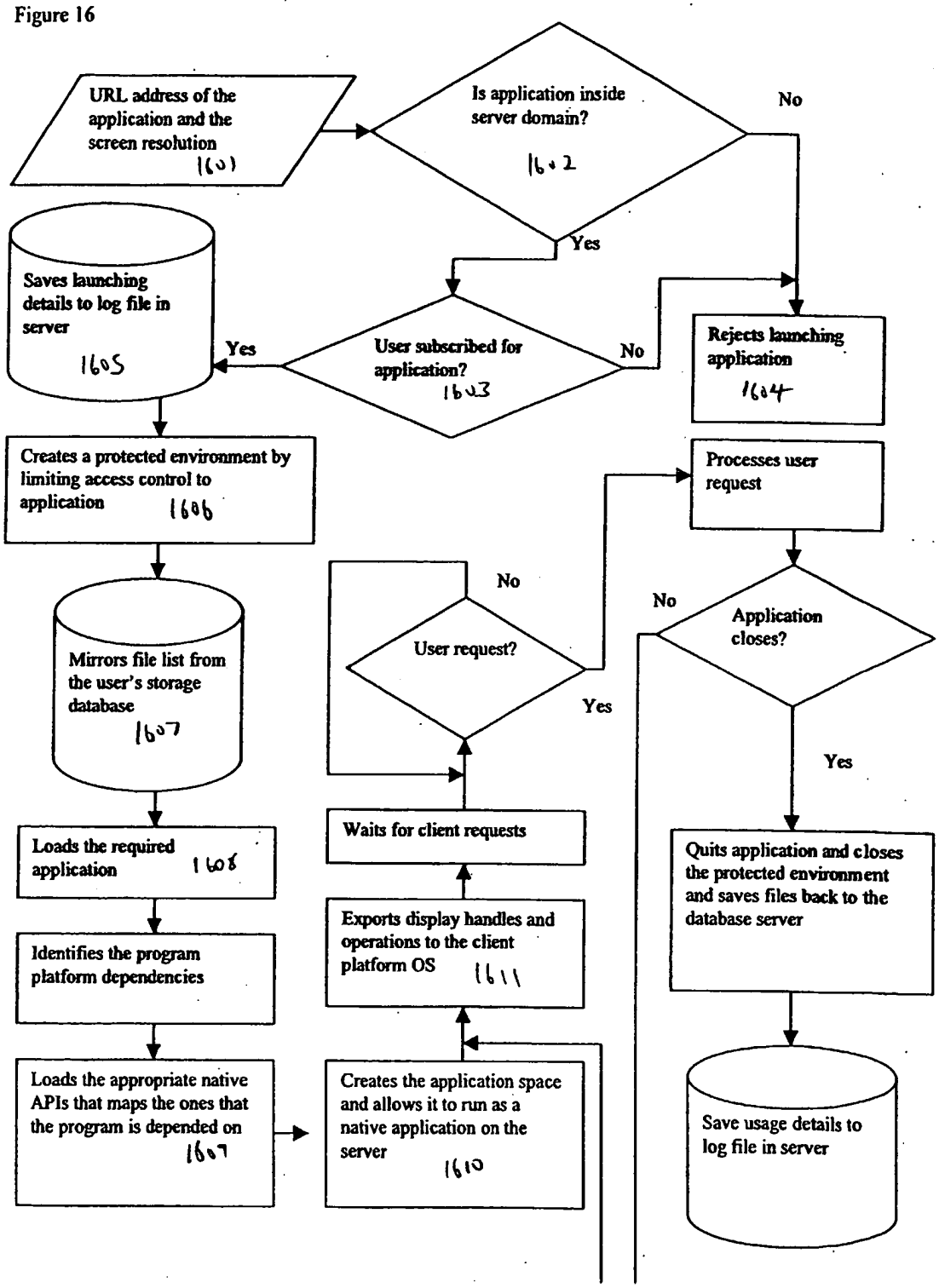

URL address of the application and the screen resolution
1601
Is application inside server domain?
1602
No
Yes
Saves launching details to log file in server
1605
Yes
User subscribed for application?
1603
No
Rejects launching application
1604
Creates a protected environment by limiting access control to application
1606
Processes user request
Mirrors file list from the user's storage database
1607
No
User request?
Yes
No
Application closes?
Yes
Loads the required application
1608
Waits for client requests
Quits application and closes the protected environment and saves files back to the database server
Identifies the program platform dependencies
Exports display handles and operations to the client platform OS
1611
Loads the appropriate native APIs that maps the ones that the program is depended on
1609
Creates the application space and allows it to run as a native application on the server
1610
Save usage details to log file in server

Figure 17

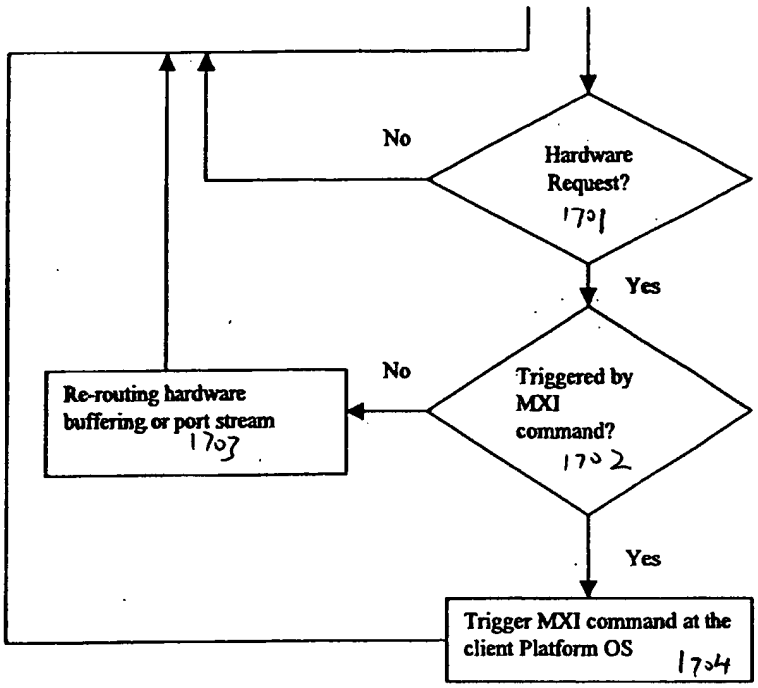

No
Hardware Request?
1701
Yes
No
Triggered by MXI command?
1702
Re-routing hardware buffering or port stream
1703
Yes
Trigger MXI command at the client Platform OS
1704

Figure 18

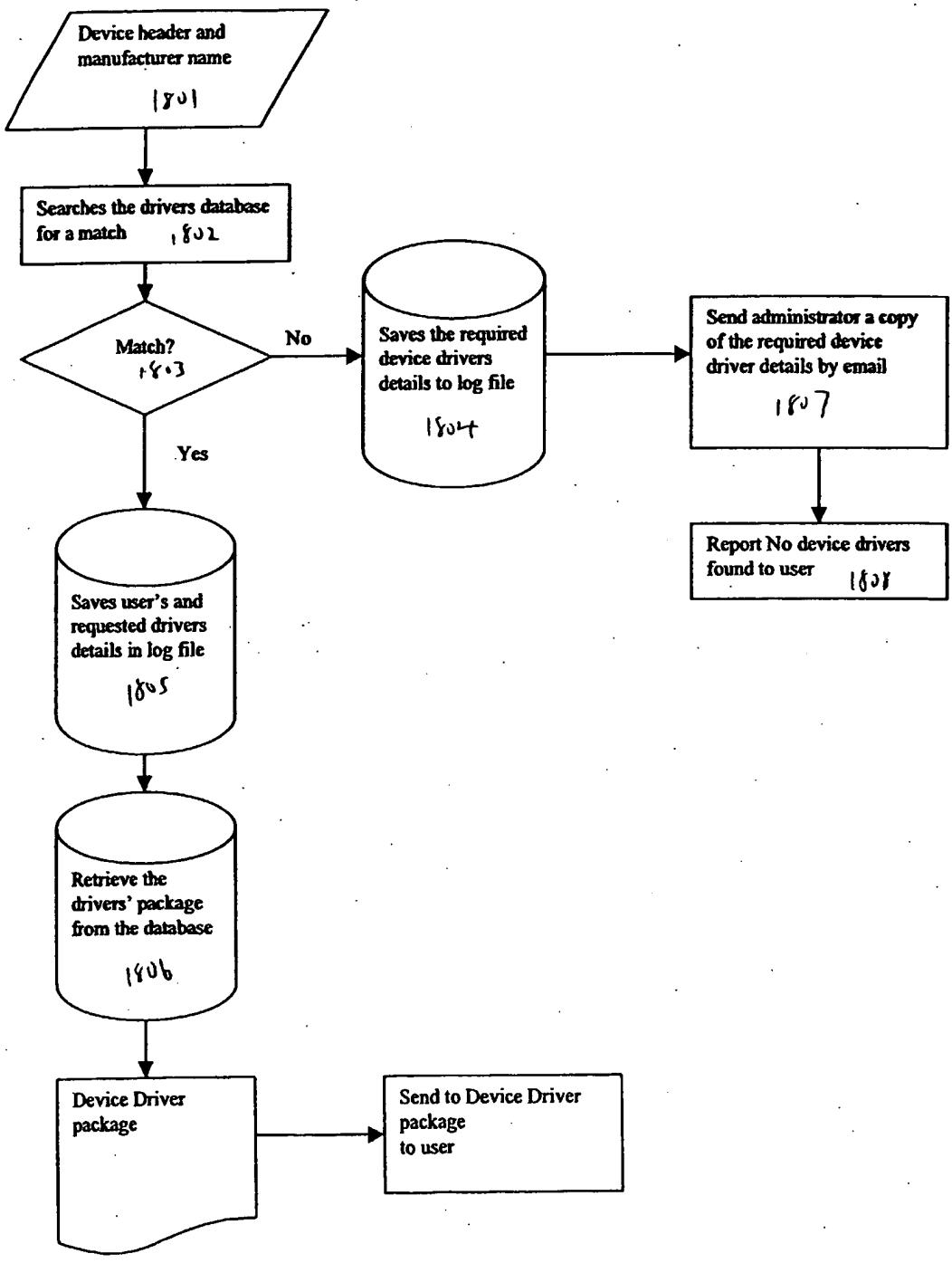

Device header and manufacturer name
1801
Searches the drivers database for a match
1802
Match?
1803
No
Yes
Saves the required device drivers details to log file
1804
Send administrator a copy of the required device driver details by email
1807
Report No device drivers found to user
1808
Saves user's and requested drivers details in log file
1805
Retrieve the drivers' package from the database
1806
Device Driver package
Send to Device Driver package to user

Figure 19

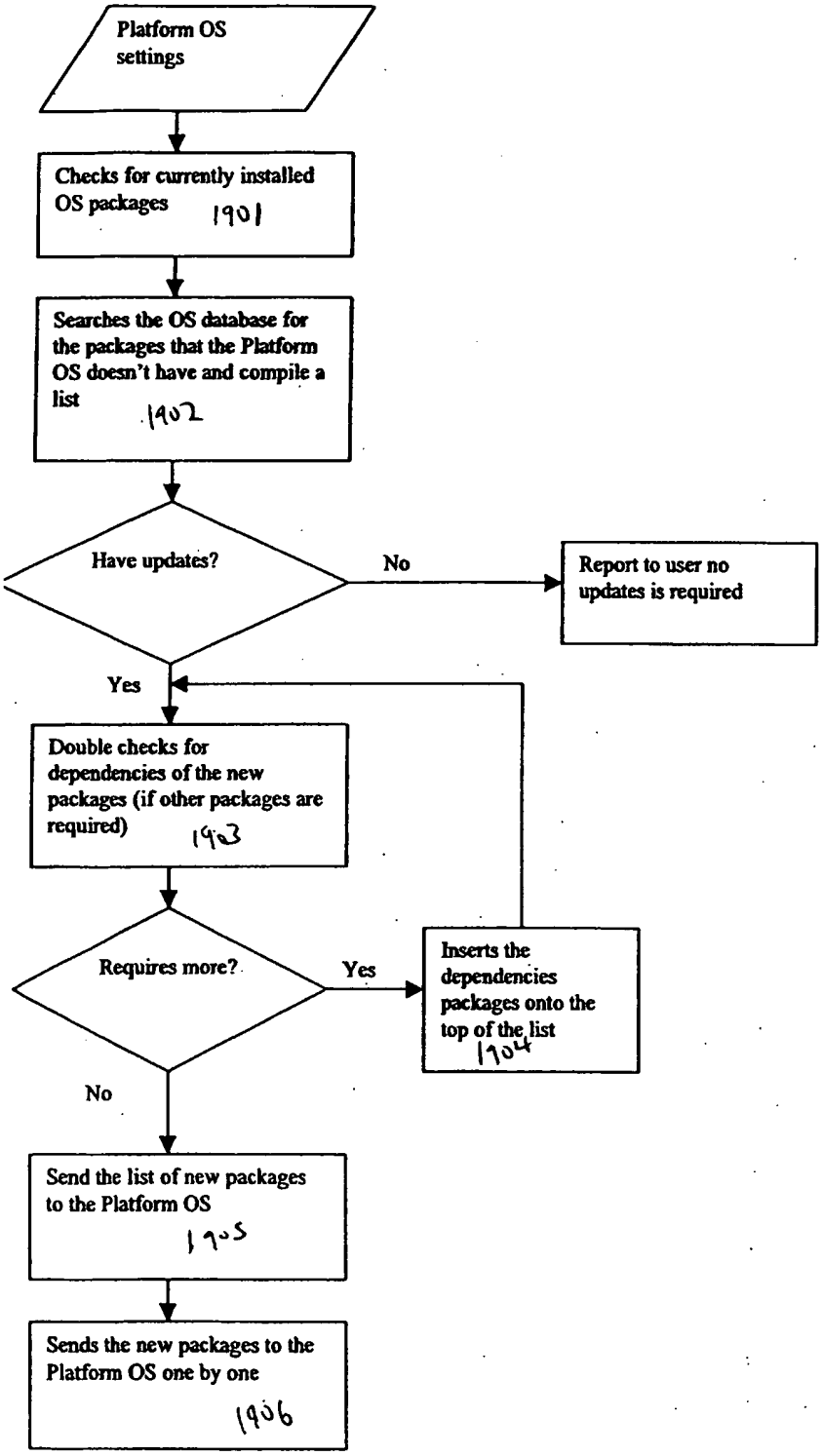

Platform OS settings
Checks for currently installed OS packages
1901
Searches the OS database for the packages that the Platform OS doesn't have and compile a list
1902
Have updates?
No
Report to user no updates is required
Yes
Double checks for dependencies of the new packages (if other packages are required)
1903
Requires more?
Yes
Inserts the dependencies packages onto the top of the list
1904
No
Send the list of new packages to the Platform OS
1905
Sends the new packages to the Platform OS one by one
1906

Figure 20

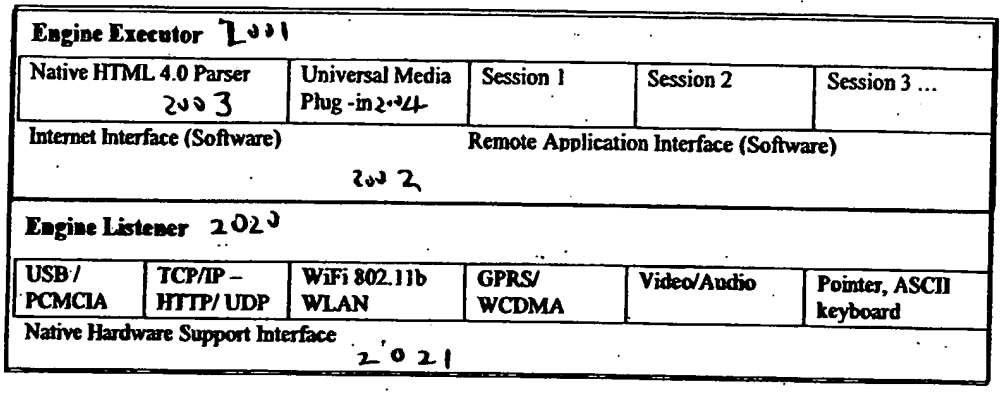


Figure 21

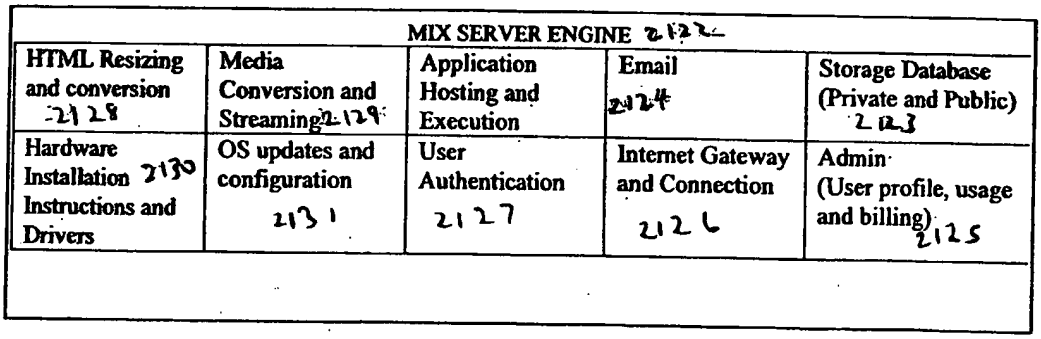
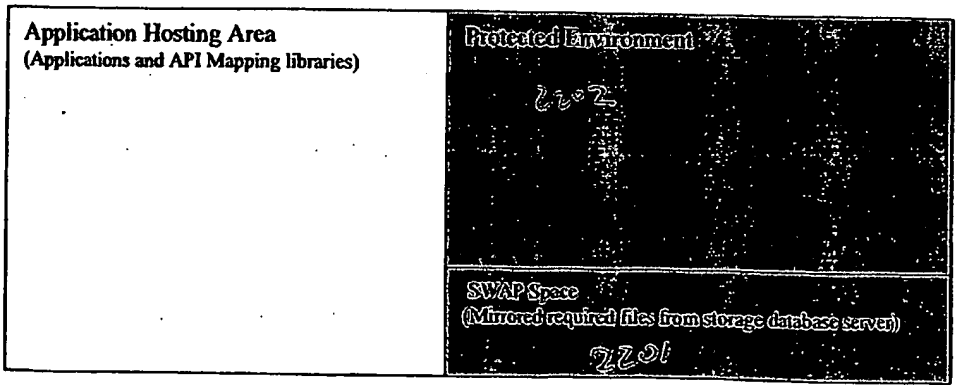


Figure 22



BEST AVAILABLE COPY